LATENTUMM: DUAL LATENT ALIGNMENT FOR UNIFIED MULTIMODAL MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unified multimodal models (UMMs) achieve strong performance in both understanding and generation by learning a shared latent space, yet they often exhibit functional inconsistency between these two capabilities. We observe that this issue does not stem from a lack of shared representations, but from the absence of explicit alignment between the transformations that map into and out of the latent space. As a result, generation and re-encoding can follow inconsistent trajectories, leading to semantic drift under modality transitions. In this work, we propose **LatentUMM**, a framework that constructs an enhanced shared latent space to explicitly align these transformations and improve cross-modal consistency. LatentUMM consists of two stages. First, dual latent alignment enforces consistency at both the modality and capacity levels: cross-modal alignment uses a stronger embedding model to impose structured cross-modal semantics, while dual capacity alignment enforces bidirectional consistency under generation and re-encoding. Second, latent dynamics stabilization improves robustness via stochastic latent rollouts and preference optimization, favoring trajectories that better preserve semantic consistency. Experiments show that LatentUMM consistently improves multimodal consistency across diverse architectures. Code is available at: `https://anonymous.4open.science/r/LatentUMM/`.

1 INTRODUCTION

Unified multimodal models (UMMs) have emerged as a promising paradigm for integrating understanding and generation within a single architecture (Zhao et al., 2025; Chen et al., 2025b; Deng et al., 2025; Xie et al., 2025). By jointly training on multimodal data, UMMs learn a shared latent space that supports both multimodal understanding and generation, enabling them to interpret inputs and produce outputs across modalities. Despite this progress, recent studies reveal a persistent inconsistency between these two capabilities (Yan et al., 2025; Wang et al., 2026a; Wen et al., 2026; Su et al., 2026b; Mao et al., 2025; Yang et al., 2026).

In particular, although UMMs can generate high-quality images, they often fail to maintain semantic consistency when processing their own outputs. For example, a model may generate an image that correctly reflects a textual prompt, yet produce mismatched or incomplete semantics when asked to reinterpret that same image (Jiang et al., 2026; Wang et al., 2026a; Han et al., 2025). This discrepancy suggests that joint training primarily aligns representations at the feature level, but does not guarantee consistency between the functional behaviors of understanding and generation.

We argue that this limitation arises because sharing a latent space is not sufficient to align how the model uses it. More fundamentally, learning a semantically consistent and cross-modal coherent latent space is highly non-trivial (Jiang et al., 2023). Unlike explicit symbolic representations, latent spaces are learned implicitly and lack direct supervision on their structure, making them difficult to interpret, constrain, or verify (Yu et al., 2026; Liu et al., 2019). As a result, semantic information is not necessarily preserved in a consistent manner across modalities, especially during transitions between understanding and generation. Although both capabilities operate on the same latent representation, they rely on different mappings—understanding process encodes inputs into the latent space, while generation decodes latent representations into outputs (Zhu et al., 2024). These mappings are learned implicitly during joint training and are not explicitly coordinated (Zhu et al., 2017; Hafner et al., 2019; Xiao et al., 2025). As a result, transitions across modalities can become inconsistent, leading to semantic drift when the model alternates between understanding and generation.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

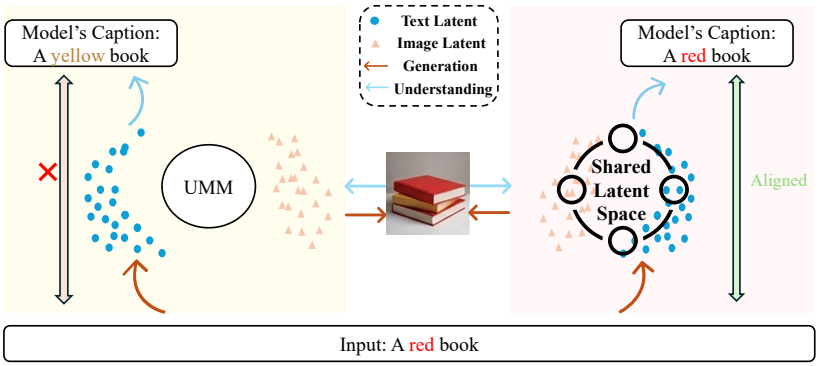


Figure 1: Overview of baseline UMMs (left) versus our proposed LatentUMM (right).

This phenomenon can be directly observed through a consistency diagnostic that measures semantic drift under repeated cross-modal transformations (details in Appendix A). Even when operating within a shared latent space, baseline UMMs exhibit progressively increasing deviation in latent representations across transformation steps, providing evidence of latent drift.

In response to this issue, a line of work on self-correction has been explored, including both inference-time refinement and post-training approaches, where models iteratively improve consistency by re-evaluating and revising their outputs (Han et al., 2025; Luo et al., 2026a; Jin et al., 2025; Mao et al., 2025; Yang et al., 2025). While effective in improving empirical performance, these methods operate by iteratively correcting outputs or adapting model behavior, without explicitly constraining the underlying interaction between understanding and generation within the shared latent space. As a result, they do not address a key source of inconsistency, i.e., the lack of alignment between the bidirectional encoding and decoding processes.

To address this issue, we propose **LatentUMM**, a framework that enforces *dual latent alignment* as a two-step process. Instead of relying solely on a shared representation space learned via joint training, our approach explicitly couples the transformations of understanding and generation through the following key steps: 1) Dual Capacity Alignment. We model both capabilities as bidirectional mappings within a shared latent space and enforce a structured alignment between their induced transformations. This ensures that information remains consistent when transitioning across modalities, such that generated outputs can be reliably reinterpreted without semantic degradation. As illustrated in Figure 1, while baseline UMMs exhibit divergence under modality loopback, LatentUMM maintains consistent structure and semantics through this alignment. 2) Latent Dynamics Stabilization. However, enforcing alignment along a single transformation path may be insufficient in complex scenarios. To further improve robustness and handle ambiguity, we introduce a rollout-based optimization mechanism in the latent space. By perturbing latent representations, we sample multiple candidate transformation trajectories and evaluate their consistency, selecting more stable paths via preference optimization. This allows the model to better handle ambiguity and avoid degenerate or inconsistent mappings before incorporating them into a post-training stage as an additional supervisory signal.

Contributions. Our contributions are three-fold:

1. **Latent alignment.** We propose LatentUMM that explicitly enforces alignment of both modalities and capacities in a shared latent space, beyond the original UMM latent space.
2. **Rollout-based latent optimization.** We introduce a rollout and preference optimization strategy that explores multiple latent transformation trajectories and selects semantically consistent ones, improving robustness and stability.
3. **Extensive experiments.** We demonstrate that LatentUMM is model-agnostic and consistently improves UMM consistency across diverse architectures.

2 Related Work

Unified Multimodal Models. Recent advances in UMMs aim to integrate multimodal understanding and generation within a single architecture (Zhao et al., 2025; Yin et al., 2024; Chen et al., 2025a; Geyer et al., 2023; Wang et al., 2026b). A dominant approach adopts decoder-only autoregressive transformers trained on interleaved multimodal tokens (Cui et al., 2025; Wang et al., 2024). Another

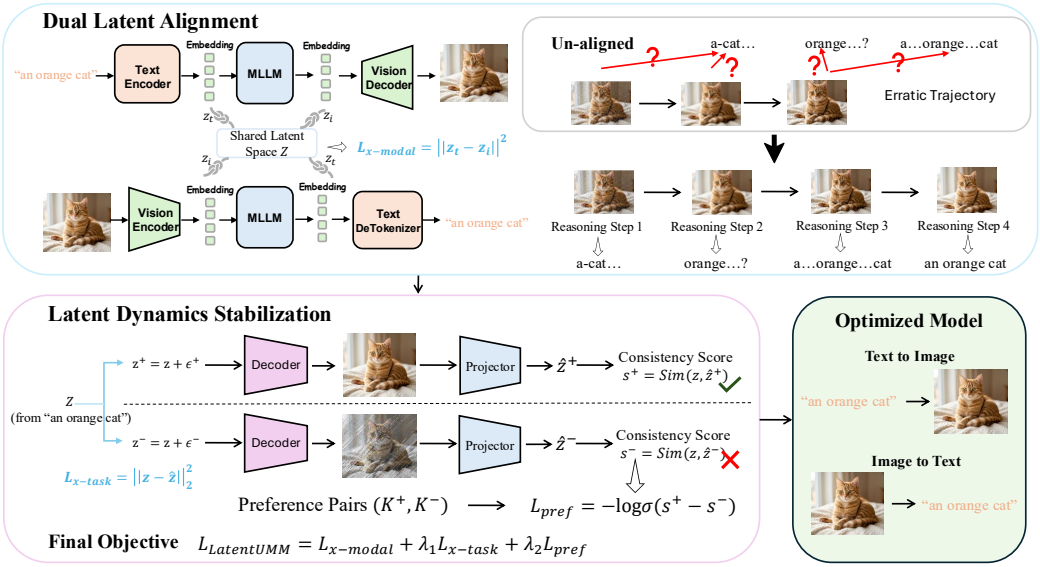


Figure 2: Overview of LatentUMM.

line of work explores hybrid generative frameworks that combine autoregressive modeling with diffusion or flow-based components (Xie et al., 2024; Wu et al., 2024; Ma et al., 2024; Chen et al., 2025b), which improve visual generation and cross-modal alignment. Additional efforts investigate modular or lightweight designs that bridge multimodal LLMs and generative models through intermediate connectors (Wu et al., 2025a). Existing approaches primarily focus on architectural unification and scaling, not cross-modal consistency and reasoning.

Inconsistency in UMMs. Recent work has begun to examine whether UMMs exhibit consistent behavior across understanding and generation (Huang et al., 2025; Qin et al., 2025; Jin et al., 2026). Despite sharing a common architecture, recent work reveals a persistent inconsistency. In particular, models that generate high-quality outputs often fail to maintain semantic consistency when processing their own generations (Wang et al., 2026a; Wen et al., 2026; Jiang et al., 2026). This indicates weak bidirectional coherence and suggests that current UMMs do not fully unify understanding and generation. Prior work attributes this issue to the nature of existing training objectives. While modalities are aligned in a shared latent space through distribution-level supervision (Radford et al., 2021; Li et al., 2023), the mappings into and out of this space are not explicitly coordinated. As a result, understanding and generation remain loosely coupled and can exhibit inconsistent behavior (Wu et al., 2024; Xie et al., 2024). Recent benchmarks further highlight this limitation by evaluating consistency under modality loopback or cross-modal transformations (Wang et al., 2026a; Yan et al., 2025; Zou et al., 2025). These results show that strong task performance does not guarantee consistent cross-modal behavior, motivating the need for explicit bidirectional alignment.

3 METHOD

UMMs are typically trained to jointly model understanding and generation within a shared latent space. While this joint training encourages a degree of cross-modal alignment, we observe a persistent *functional inconsistency* between the two capabilities: representations that support strong multimodal understanding do not always induce faithful generation, and conversely, generative trajectories often deviate from semantically consistent latent semantics (Rohrbach et al., 2018; Chen et al., 2021; Ho & Salimans, 2022).

This suggests that a single shared latent space optimized only under joint training is insufficient to guarantee cross-capability consistency. To address this limitation, we propose to construct an **enhanced shared latent space** that is explicitly refined using a stronger embedding model. The key idea is to further align the latent representations induced by understanding and generation, such that both functions become mutually consistent under the same semantic geometry.

Overview. We propose LatentUMM, a framework that refines the latent space of a pretrained UMM into a more consistent multimodal representation space. As shown in Figure 2, LatentUMM operates as a two-step process: (1) Dual Latent Alignment leverages a stronger embedding model to regularize latent semantics and enforces consistency between understanding-induced and generation-induced transformations. (2) Latent Dynamics Stabilization then applies rollout-based preference optimization to handle complex scenarios, ensuring robust and stable latent trajectories.

3.1 Problem Formulation

Given paired text-image data (x_t, x_i), where x_t and x_i denote text and image inputs respectively, we aim to learn a unified latent representation $z \in \mathcal{Z}$ in a shared latent space $\mathcal{Z}$. We define modality-specific encoders $E_t(\cdot)$ and $E_i(\cdot)$ that map inputs into modality-specific latent embeddings: $z_t = E_t(x_t)$ and $z_i = E_i(x_i)$, where we assume $z_t, z_i \in \mathbb{R}^d$, i.e., they lie in the same latent dimensionality d for compatibility in subsequent fusion (Radford et al., 2021; Faghri et al., 2017; Frome et al., 2013). We construct the unified latent representation via a fusion operator $F(\cdot, \cdot)$: $z = F(z_t, z_i)$, where F is implemented by UMM's latent fusion module (e.g., cross-attention or pooling), and is kept fixed during our refinement stage unless otherwise specified. A decoder $G(\cdot)$ maps latent representations back to the image space: $\hat{x} = G(z)$.

To construct a semantically more stable supervision signal over the same latent dimensionality, we use a stronger embedding model $E^*(\cdot)$ to map any input into a refined embedding space: $\phi(x) = E^*(x) \in \mathbb{R}^d$, where we explicitly assume that E^* produces embeddings in the same dimensionality d, enabling direct geometric comparison (i.e, gemini embedding model (Lee et al., 2025) projects all modalities into same dimension). Although dimensionality reduction or projection into a different space is common, it would require an additional learned mapping (Radford et al., 2021; Chen et al., 2020), potentially introducing optimization complexity and obscuring the source of alignment improvements. Our design isolates the role of the supervision signal itself. For convenience, we define: $\phi(\hat{x}) := E^*(\hat{x})$. Here, E^* is a fixed high-capacity model used only for supervision and alignment but not for inference. LatentUMM aims to perform dual latent alignment to improve consistency, which will be introduced in next sections.

3.2 Analysis of Distributional Alignment in Latent Space

We formalize how an auxiliary cross-modal semantic space can reduce the distributional gap between text and image representations. Let $Z_t \sim P_t$ and $Z_i \sim P_i$ denote the original representations, and let $A_t \sim Q_t$ and $A_i \sim Q_i$ denote normalized or projected representations in the auxiliary space. We measure the gap with the 2-Wasserstein metric:

$$W_2(P, Q) = \inf_{(U,V)\in\Pi(P,Q)} \left(\mathbb{E}\left[\|U - V\|_2^2\right]\right)^{1/2}, \tag{1}$$

where $\Pi(P, Q)$ is the set of couplings with marginals P and Q. Define $\delta = W_2(P_t, P_i)$, and $\varepsilon = W_2(Q_t, Q_i)$. The auxiliary space is stronger when $\varepsilon < \delta$. Consider an alignment update toward the auxiliary representations:

$$\widetilde{Z}_t = (1-\alpha)Z_t + \alpha A_t, \qquad \widetilde{Z}_i = (1-\alpha)Z_i + \alpha A_i, \qquad 0 \le \alpha \le 1. \tag{2}$$

The resulting representations induce distributions $\widetilde{P}_t$ and $\widetilde{P}_i$, respectively, which satisfy the following bound that guarantees smaller cross-modal distribution gap (proof in Appendix B):

$$W_2(\widetilde{P}_t, \widetilde{P}_i) \le (1-\alpha)\delta + \alpha\varepsilon. \tag{3}$$

3.3 Dual Latent Alignment

Dual latent alignment aims to achieve two levels of alignment: dual modal alignment between the visual and textual modalities, and dual capacity alignment for both the understanding and generation capacities of UMMs by modeling them in a shared latent space. A stronger embedding model E^* is used to induce both image and text representations into the same latent space that facilitates cross-modal alignment explicitly. Then, we can perform capability alignment to improve the consistency.

Dual modal alignment. We align trainable UMM representations with the semantic targets induced by E^*. Let $u_t = P_t(z_t)$ and $u_i = P_i(z_i)$ denote the corresponding alignment-head outputs, and let

$a_t = R\phi(x_t)$ and $a_i = R\phi(x_i)$ denote normalized auxiliary targets. We use:

$$\mathcal{L}_{\text{x-modal}} = \frac{1}{2}\|u_t - a_t\|_2^2 + \frac{1}{2}\|u_i - a_i\|_2^2 + \gamma\|u_t - u_i\|_2^2. \tag{4}$$

Here, P_t, P_i, and R are identity maps when all representations already share the same normalized coordinates. This objective supplies a gradient to the UMM alignment heads while enforcing both anchor and cross-modal alignment. In real experiments, the embedding model E^* has many options such as Gemini Embedding model (Lee et al., 2025) and SigLIP (Zhai et al., 2023), whose effectiveness is ablated in §4.3.1.

Dual capacity alignment. Based on the alignment in shared space, to further improve consistency between understanding and generation, we define a bidirectional process in latent space. Starting from a unified latent representation z, we generate an output $\hat{z} = \phi(\hat{x})$ and re-encode it into the refined embedding space: $z \xrightarrow{G} \hat{x}$. We then enforce latent bidirectional alignment to ensure that generation and re-encoding preserve semantic identity in the refined latent space: $\mathcal{L}_{\text{x-task}} = \|z - \hat{z}\|_2^2$.

3.4 Latent Dynamics Stabilization

However, these latent alignments are only performed at instance level, lacking distributional robustness for diverse samples in real-world applications. Latent dynamics stabilization is further performed to improve the robustness. The core idea is to leverage stochastic latent rollouts to generate perturbations, which will then be used to compute the similarity of each trajectory. However, stochastic rollouts inevitably introduce high-variance perturbations, resulting in latent trajectories with varying semantic fidelity. While some trajectories preserve the underlying semantics, others may drift due to compounding generation and re-encoding errors. This naturally induces a relative ranking over trajectories based on their semantic consistency. A straightforward approach would be to directly regress on these scores or average across trajectories. However, such objectives are sensitive to noisy or low-quality samples and may blur the semantic structure of the latent space. Instead, we formulate latent dynamics stabilization as a preference learning problem, which focuses on distinguishing more consistent trajectories from less consistent ones. This relative supervision provides a more robust and discriminative training signal under stochastic perturbations.

Stochastic latent rollouts. We first sample stochastic perturbations around latent representation:

$$z^{(k)} = z + \epsilon^{(k)}, \quad \epsilon^{(k)} \sim \mathcal{N}(0, \sigma^2 I), \quad k = 1, \ldots, K, \tag{5}$$

where $\epsilon^{(k)}$ denotes Gaussian noise sampled for the k-th rollout, σ controls the perturbation magnitude (ablation in §E.1), K is the number of sampled trajectories whose efficacy is studied in §4.3.2. Each trajectory follows: $z^{(k)} \to \hat{x}^{(k)} = G(z^{(k)}) \to \hat{z}^{(k)} = \phi(\hat{x}^{(k)})$. We then define a similarity function in the refined embedding space: $\text{Sim}(a, b) = \frac{a^\top b}{\|a\|\|b\|}$. As such, the self-consistency score of each trajectory is: $s^{(k)} = \text{Sim}(z, \hat{z}^{(k)})$.

Preference optimization. Following preference-based optimization (Rafailov et al., 2023), denote $\sigma(\cdot)$ as the sigmoid function, the preference loss is defined as:

$$\mathcal{L}_{\text{pref}} = -\log \sigma\big(s^{(k^+)} - s^{(k^-)}\big), \tag{6}$$

where the preference pair (k^+, k^-) is defined as: $k^+ = \arg\max_k s^{(k)}$, and $k^- = \arg\min_k s^{(k)}$. Then, the final training objective for LatentUMM is formulated as:

$$\mathcal{L}_{\text{LatentUMM}} = \mathcal{L}_{\text{x-modal}} + \lambda_1 \mathcal{L}_{\text{x-task}} + \lambda_2 \mathcal{L}_{\text{pref}}, \tag{7}$$

where $\lambda_1, \lambda_2 > 0$ are two trade-off hyperparameters that control the relative importance of dual consistency and preference optimization, respectively.

LatentUMM thus constructs a refined shared latent space that enforces consistency between understanding and generation via both cross-modal alignment and bidirectional-consistent latent dynamics. Unlike standard UMM training that relies solely on joint optimization, our method introduces an external embedding-driven geometric constraint that stabilizes multimodal reasoning trajectories.

4 Experiments

4.1 Experimental Setup

Datasets. We perform training on Text-to-Image-2M dataset (Xie et al., 2026), a large-scale collection of paired text-image data. We conduct evaluation across generation, understanding, and editing

Table 1: Results on multimodal understanding benchmarks.

Model	Params	MME	MMMU	MMVet	MMBench-EN	MMBench-CN	MathVista	
							Free-Form	Overall
Bagel	14B	1691.4	51.9	65.9	82.65	80.94	61.52	71.3
Janus Pro	7B	1547.9	40.7	49.3	64.84	64.94	32.83	42.8
Janus	1.3B	1221.4	27.3	27	27.69	37.19	16.3	26.6
Janus Flow	1.3B	1305.6	29	31.8	64.16	60.31	29.78	34.8
Show-O2	7B	1619.8	47.9	47.1	77.25	76.68	37.39	51.5
Show-O2	1.5B	1413.3	37.1	46.1	65.09	61.04	19.78	37.9
Show-O	1.3B	1188.5	26.1	23.3	43.48	11.1	11.3	29
Emu3	8B	1176	31.4	30	59.53	45.66	30	44.9
Emu3.5	34B	832.2	29.2	28	17.56	18.13	17.61	41.7
Omnigen2	7B	1584.4	46	62.7	76.36	75.43	53.91	63.5
MMaDA	8B	939	28.9	11.4	28.16	20.57	8.7	24.9
Bagel+SFT	–	1690.1	51.9	65.8	82.65	80.83	65.43	73.1
Bagel+RecA	–	1689.1	52.3	66.1	82.65	80.94	64.57	51.6
Bagel+UniGame	–	1692.1	52.4	60.7	82.75	80.94	63.48	72.7
Bagel+UniCot	–	1690.5	53.1	64.5	83.12	80.99	64.13	73
Bagel+LatentUMM	–	**1695.8**	**54.1**	**67.2**	**84.99**	**84.91**	**65.45**	**73.6**

Table 2: Results on multimodal generation benchmarks.

	DPG-Bench						UEval			WISE
	Global	Entity	Attribute	Relation	Other	Overall	Text	Image	Overall	
Bagel	82.07	90.22	87.82	93.26	82.26	84.1	54.97	6.84	30.9	0.399
+SFT	81.25	89.56	87.48	93.57	82.40	83.78	55.56	6.65	31.1	0.407
+LatentUMM	82.37	91.34	88.88	93.58	88.8	85.62	55.38	8.23	31.8	0.418

tasks. Generation quality is measured using DPG-Bench (Hu et al., 2024), U-Eval (Li et al., 2026), and WISE (Niu et al., 2025). Understanding is evaluated on MME (Fu et al., 2023), MMMU (Yue et al., 2024), MMBench (Liu et al., 2024), MathVista (Lu et al., 2023) and MM-Vet (Yu et al., 2023). Editing is evaluated on ImgEdit benchmark (Ye et al., 2025). Furthermore, we evaluate the consistency on Unified-Bench (Yan et al., 2025) and RealUnify (Shi et al., 2025). All experiments are conducted using TorchUMM (Luo et al., 2026b) for fair comparison.

Backbones and Embedding Models. Bagel (Deng et al., 2025) is selected as our main base model; Janus-Pro (Chen et al., 2025b) and Harmon (Wu et al., 2025b) are adopted for flexibility. All post-training methods, including SFT, RecA (Xie et al., 2026), UniGame (Su et al., 2026a) and UniCot (Qin et al., 2025), are trained on the same Text-to-Image-2M dataset for fair comparison. To enable alignment between text and image representations, we employ external embedding models to map both modalities into a shared latent space. Specifically, we extract embeddings for text and images using a pretrained encoder and project them into a unified representation space with the same dimensionality. By default, we use the Gemini embedding model (Lee et al., 2025). We further perform ablations with CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023).

Efficiency and complexity. Detailed complexity analysis and empirical efficiency measurements are provided in Appendix Section D.

4.2 Main Results

Improved understanding performance. As shown in Table 1, LatentUMM consistently improves understanding and outperforms post-training baselines on all benchmarks. The improvements are most evident on comprehensive evaluation suites (MME and MMVet), where LatentUMM achieves the strongest gains, suggesting that latent consistency effectively enhances global multimodal alignment rather than overfitting to specific task formats. Notably, LatentUMM achieves the best performance on open-ended reasoning (MathVista Free-Form), indicating stronger capability in handling less structured, generative reasoning tasks. Compared to UniCot, whose chain-of-thought-style supervision appears to benefit structured reasoning benchmarks, LatentUMM 's latent consistency constraint better supports flexible reasoning that requires integrating generation and understanding.

Improved generation performance. In Table 2, LatentUMM consistently outperforms the baseline model on all generation benchmarks. On DPG-Bench, the gains are not only reflected in the overall score but are particularly pronounced in fine-grained dimensions such as entity and attribute gener-

ation, indicating improved precision in capturing object details and their properties. These results suggest that enforcing latent consistency improves both the fidelity of fine-grained content generation and the robustness of multimodal outputs, leading to more reliable and coherent generation across diverse scenarios.

Improved editing performance. LatentUMM also brings consistent improvements on the editing benchmark (Table 3). On the overall setting, the gains are most pronounced in geometric mean, reflecting a balanced improvement across both semantic fidelity and visual quality. These results indicate that enforcing latent consistency enhances both the semantic accuracy and perceptual quality of edits, while improving their joint alignment. This leads to more stable and reliable editing outcomes, especially in scenarios requiring precise, localized modifications.

Table 3: Results on editing benchmarks.

Model	Overall			Intersection		
	SC	PQ	O	SC	PQ	O
Bagel	6.679	7.004	6.348	6.726	7.027	6.384
+LatentUMM	6.853	7.009	6.518	6.987	7.037	6.634

4.3 Ablation Study

4.3.1 Effect of Shared Latent Space and Embedding Models

Shared Latent Space. We first evaluate the importance of shared latent space by comparing against a direct SFT baseline, which does not impose any alignment constraint. As shown in Tables 1 and 2, while SFT remains competitive on certain metrics, it consistently underperforms alignment-based variants in overall performance. This comparison highlights the benefit of explicitly modeling cross-modal consistency. By aligning generation and understanding within a shared representation space, the model achieves more coherent and transferable representations, leading to improved generalization across tasks. Moreover, Table 4(a) compares two strategies for leveraging the shared latent space. Alignment within the existing shared latent space directly regularizes the pretrained latent representation, while constructing and enhanced shared latent space reshapes the latent geometry under the guidance of a stronger embedding model. We observe that our method consistently achieves better performance, suggesting that the original latent space of the pretrained UMM is not sufficiently structured for reliable cross-modal alignment. Simply enforcing alignment within this space is therefore limited by its inherent geometric imperfections.

Embedding Models. Table 4(b) shows that replacing Gemini embedding with CLIP or SigLIP leads to slightly different performance trade-offs, and no single approach dominates. The performance gap across embedding choices remains relatively small, indicating that LatentUMM is *not*

Table 4: Ablation study on different aspects of LatentUMM: (a) shared latent space, (b) embedding model, (c) rollout strategy, (d) noise scale, and (e) decoding strategy .

Model	MME	MMMU	MMVet	MMBench-EN	MMBench-CN	MathVista	
						Free-Form	Overall
(a) Effect of Shared Latent Space							
Alignment within Existing Shared Latent Space	1691.5	52.3	66.4	83.17	80.52	62.72	71.7
Alignment through Added Shared Latent Space	1695.8	54.1	67.2	84.99	84.91	65.45	73.6
(b) Embedding Model							
Bagel	1691.4	51.9	65.9	82.65	80.94	61.52	71.3
CLIP	1694.4	52.8	67.2	83.17	82.84	62.28	73.3
SigLIP	1693.7	54.2	66.7	84.53	82.46	65.12	73.4
Gemini Embedding Model (Ours)	1695.8	54.1	67.2	84.99	84.91	65.45	73.6
(c) Rollout Strategy (based on Gemini embeddings)							
Rollout step = 5	1697.3	52.8	66.1	83.98	80.82	65.10	73.2
Rollout step = 10 (default)	1695.8	54.1	67.2	84.99	84.91	65.45	73.6
Rollout step = 20	1693.1	52.3	66.7	85.11	80.98	65.72	73.5
(d) Noise Scale (in latent sampling)							
Noise = 0.0 (deterministic)	1689.7	52.0	65.9	83.45	80.65	64.80	72.9
Noise = 0.1 (default)	1695.8	54.1	67.2	84.99	84.91	65.45	73.6
Noise = 0.2	1692.6	54.1	67.3	85.37	82.90	65.40	73.3
(e) Inference (Decoding) Strategies							
Single-pass decoding	1688.9	51.9	65.8	80.16	80.60	64.90	72.9
Simple ensembling	1692.7	52.3	66.3	81.78	80.95	65.50	73.4
Self-consistency decoding	1695.8	54.1	67.2	84.99	84.91	65.45	73.6

Table 5: Results on unification benchmarks for consistency evaluation.

Model	UniSon			Unified-Bench					RealUnify (GEU)				
	Int. Und.	Int. Gen.	Int. Uni.	CLIP	DINOv2	DINOv3	LongCLIP	Overall	MC	CN	AF	MR	Total
Bagel	96.1	82.6	80.5	0.8947	0.7877	0.7240	0.9321	0.8346	0.3	0.39	0.52	0.34	0.3875
+SFT	96.9	82.4	80.2	0.8922	0.7809	0.7203	0.9351	0.8321	0.32	0.32	0.47	0.34	0.3625
+Unicot	97.1	82.7	80.9	0.9026	0.7774	0.7128	0.9392	0.8330	0.32	**0.40**	0.5	**0.35**	0.3925
+LatentUMM	**97.6**	**83.7**	**82.1**	**0.9102**	**0.7988**	**0.7341**	**0.9417**	**0.8462**	**0.32**	**0.40**	**0.55**	**0.35**	**0.4050**

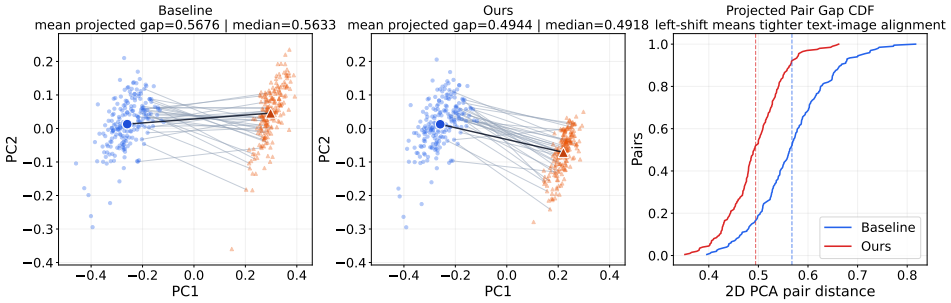


Figure 3: Latent space analysis

tightly coupled to a specific embedding model. This suggests that the gains primarily arise from the alignment mechanism rather than the embedding model.

4.3.2 Effect of Rollout and Inference Design

We further analyze the impact of rollout depth, stochasticity, and decoding strategies.

Rollout steps. Different rollout frequency leads to non-uniform effects across benchmarks. Shorter rollouts (e.g., $K = 5$) improve certain metrics such as MME and MMMU, while longer rollouts (e.g., $K = 20$) slightly benefit free-form reasoning on MathVista. However, no single configuration dominates, indicating a trade-off between stability and long-horizon aggregation (Table 4(c)).

Noise scale. Introducing moderate stochasticity improves performance compared to deterministic inference (Table 4(d)). In particular, a small noise level (0.1) provides a balance between exploration and stability, while excessive noise (0.2) begins to degrade on structured benchmarks.

Decoding strategy. Advanced decoding strategies further improve robustness (Table 4(e)). Both simple ensembling and self-consistency outperform single-pass decoding, with self-consistency yielding the most stable gains. However, the improvements remain incremental, suggesting that decoding primarily refines predictions rather than fundamentally altering model behavior.

4.4 Analysis on Consistency Improvement

Consistency evaluation on unification benchmarks. We evaluate consistency on Unified-Bench (Yan et al., 2025) and RealUnify (Shi et al., 2025) that jointly measures generation quality and re-interpretation consistency within a single pipeline. As shown in Table 5, LatentUMM consistently outperforms the base model and SFT baseline across unified metrics, with particularly notable gains in consistency-oriented evaluation. This indicates that the model not only generates high-quality outputs but also maintains stronger semantic alignment when re-interpreting them. In contrast, SFT achieves competitive results on individual tasks but underperforms in the unified setting, highlighting the limitation of optimizing generation and understanding independently.

Table Table 5 also reports the UniSon internal scores (Liu et al., 2026), which measure consistency within the understanding and generation pathways. On the Bagel backbone, LatentUMM improves internal understanding from 96.1 to 97.6, internal generation from 82.6 to 83.7, and internal unification from 80.5 to 82.1. These gains provide complementary evidence that LatentUMM strengthens the shared latent representations underlying both capabilities.

Analysis of latent space alignment. Figure 3 presents the 2D PCA projections of these embeddings for both the base and our models, alongside the Cumulative Distribution Function (CDF) of the projected pair gaps. It is shown that the base model exhibits a significant dispersion between the text and image representations, characterized by a mean projected gap of 0.5676. LatentUMM reduces it to 0.4944. Furthermore, the CDF plot in the right panel confirms this tightening, where the

Table 6: Generalization across unified multimodal models.

Model	Generation			Understanding			
	DPG	WISE	UEval	MME	MMMU	MMVet	MathVista
Bagel	84.1	0.399	30.9	1691.4	51.9	65.9	71.3
+LatentUMM	85.5 (+1.4)	0.418 (+0.019)	31.8 (+0.9)	1695.8 (+4.4)	54.1 (+2.2)	67.2 (+1.3)	73.6 (+2.3)
Janus-Pro	83.73	0.381	20.6	1547.9	40.7	49.3	42.8
+LatentUMM	85.1 (+1.37)	0.403 (+0.022)	21.1 (+0.5)	1551.3 (+3.4)	41.2 (+0.5)	50.5 (+1.2)	43.9 (+1.1)
Harmon	81.19	0.379	27.9	1224.9	35.0	26.2	35.2
+LatentUMM	85.74 (+4.55)	0.392 (+0.013)	29.8 (+1.9)	1251.4 (+26.5)	37.6 (+2.6)	27.1 (+0.9)	36.8 (+1.6)

distribution for LatentUMM consistently shifts to the left compared to the baseline. This indicates that our alignment objective effectively reduces the distance between modalities, ensuring a highly coherent and tightly coupled latent structure that bridges functional inconsistency. In contrast, the baseline model shows a wider gap between modalities, highlighting the limitation of joint training without explicit alignment.

Case study. We analyze a multi-step sequential interaction task in Figure 4. The target input consists of a complex scene description involving four sequential operations: setting a transparent glass, placing a silver spoon inside, positioning a yellow napkin on the rim, and placing a red dice on the napkin and inside the glass. As shown on left, the base model generates the correct visual representation for the sequence; however, when re-evaluating its own generated image through the understanding pathway, it fails to preserve the sequential logic and spatial configuration. The model's understanding output diverges from the original prompt, incorrectly reordering the steps (e.g., placing the dice inside before the napkin covers the rim). In contrast, as shown in the right panel, LatentUMM aligns the understanding and generation pathways within the shared latent space. By leveraging the dual capability alignment and rollout-based optimization, the framework maintains the temporal and spatial dependencies across both modalities. As a result, the re-interpreted caption remains completely faithful to the original input prompt, accurately preserving the four-step action sequence without functional inconsistency.

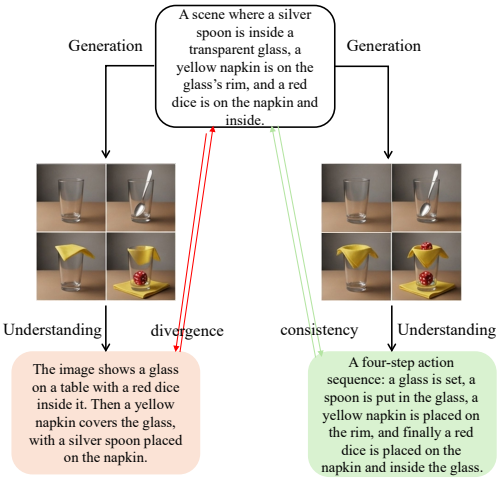


Figure 4: Baseline UMMs (left) vs. LatentUMM (right) on a sequential interaction task.

4.5 Generalization Across Backbones

We extend our experiments beyond Bagel to include Janus-Pro (Chen et al., 2025b) and Harmon (Wu et al., 2025b). As shown in Table 6, LatentUMM consistently improves both generation and understanding performance across all evaluated backbones, demonstrating that the proposed latent-space reasoning framework is not architecture-specific and can be seamlessly integrated into heterogeneous multimodal models. Further interpretation in Appendix Section E.10.

5 Conclusion

In this work, we identify a key limitation of unified multimodal models: the lack of consistency between understanding and generation despite sharing a common latent space. To address this, we propose LatentUMM, which explicitly enforces dual alignment within a refined shared latent space. Extensive experiments across generation, understanding, editing, and unified evaluation benchmarks demonstrate that LatentUMM consistently outperforms strong baselines and post-training methods, additionally, generalizes across diverse model architectures. Overall, our results suggest that achieving true multimodal unification requires not only shared representations, but also structured coordination of how these representations are used across capabilities. We hope this work provides a step toward more reliable and coherent multimodal reasoning, and offers insights for future research on aligning latent dynamics in unified models. We discuss limitations in Appendix C.

AI USE STATEMENT

We used generative AI tools to assist with manuscript revision, literature search, and code development. The experimental design and methodological choices were made by the authors.

REPRODUCIBILITY STATEMENT

Dataset protocols, preprocessing, model pools, and training hyperparameters are provided in Appendix E.

REFERENCES

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025a.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. Improving faithfulness in abstractive summarization with contrast candidate generation and selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5935–5941, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025b.

Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, et al. Emu3. 5: Native multimodal models are world learners. *arXiv preprint arXiv:2510.26583*, 2025.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. *Advances in neural information processing systems*, 26, 2013.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Yujin Han, Hao Chen, Andi Han, Zhiheng Wang, Xinyu Liu, Yingya Zhang, Shiwei Zhang, and Difan Zou. Turning internal gap into self-improvement: Promoting the generation-understanding unification in mllms. *arXiv preprint arXiv:2507.16663*, 2025.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

Wenxuan Huang, Shuang Chen, Zheyong Xie, Shaosheng Cao, Shixiang Tang, Yufan Shen, Qingyu Yin, Wenbo Hu, Xiaoman Wang, Yuntian Tang, et al. Interleaving reasoning for better text-to-image generation. *arXiv preprint arXiv:2509.06945*, 2025.

Hongbo Jiang, Jie Li, Yunhang Shen, Pingyang Dai, Xing Sun, Haoyu Cao, and Liujuan Cao. Can unified generation and understanding models maintain semantic equivalence across different output modalities? *arXiv preprint arXiv:2602.23711*, 2026.

Qian Jiang, Changyou Chen, Han Zhao, Liqun Chen, Qing Ping, Son Dinh Tran, Yi Xu, Belinda Zeng, and Trishul Chilimbi. Understanding and constructing latent modality structures in multimodal representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7661–7671, 2023.

Jiachun Jin, Zetong Zhou, Xiao Yang, Hao Zhang, Pengfei Liu, Jun Zhu, and Zhijie Deng. Latentum: Unleashing the potential of interleaved cross-modal reasoning via a latent-space unified model. *arXiv preprint arXiv:2604.02097*, 2026.

Weiyang Jin, Yuwei Niu, Jiaqi Liao, Chengqi Duan, Aoxue Li, Shenghua Gao, and Xihui Liu. Srum: Fine-grained self-rewarding for unified multimodal models. *arXiv preprint arXiv:2510.12784*, 2025.

Jinhyuk Lee, Feiyang Chen, Sahil Dua, Daniel Cer, Madhuri Shanbhogue, Iftekhar Naim, Gustavo Hernández Ábrego, Zhe Li, Kaifeng Chen, Henrique Schechter Vera, et al. Gemini embedding: Generalizable embeddings from gemini. *arXiv preprint arXiv:2503.07891*, 2025.

Bo Li, Yida Yin, Wenhao Chai, Xingyu Fu, and Zhuang Liu. Ueval: A benchmark for unified multimodal generation. *arXiv preprint arXiv:2601.22155*, 2026.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Jinyu Liu, Xincheng Shuai, Henghui Ding, and Yu-Gang Jiang. Unison: Benchmarking unified multimodal models via synergistic understanding and generation. In *International Conference on Machine Learning (ICML)*, 2026.

Ninghao Liu, Mengnan Du, and Xia Hu. Representation interpretation with spatial encoding and multimodal analytics. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 60–68, 2019.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

Yinyi Luo, Hrishikesh Gokhale, Marios Savvides, Jindong Wang, and Shengfeng He. Self-corrected image generation with explainable latent rewards. *arXiv preprint arXiv:2603.24965*, 2026a.

Yinyi Luo, Wenwen Wang, Hayes Bai, Hongyu Zhu, Hao Chen, Pan He, Marios Savvides, Sharon Li, and Jindong Wang. Torchumm: A unified multimodal model codebase for evaluation, analysis, and post-training. *arXiv preprint arXiv:2604.10784*, 2026b.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai yu, Liang Zhao, Yisong Wang, Jiaying Liu, and Chong Ruan. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation, 2024.

Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning. *arXiv preprint arXiv:2505.23380*, 2025.

Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Kunpeng Ning, Bin Zhu, et al. Wise: A world knowledge-informed semantic evaluation for text-to-image generation. *arXiv preprint arXiv:2503.07265*, 2025.

Luozheng Qin, Jia Gong, Yuqing Sun, Tianjiao Li, Mengping Yang, Xiaomeng Yang, Chao Qu, Zhiyu Tan, and Hao Li. Uni-cot: Towards unified chain-of-thought reasoning across text and vision. *arXiv preprint arXiv:2508.05606*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4035–4045, 2018.

Yang Shi, Yuhao Dong, Yue Ding, Yuran Wang, Xuanyu Zhu, Sheng Zhou, Wenting Liu, Haochen Tian, Rundong Wang, Huanqian Wang, et al. Realunify: Do unified models truly benefit from unification? a comprehensive benchmark. *arXiv preprint arXiv:2509.24897*, 2025.

Zhaolong Su, Wang Lu, Hao Chen, Sharon Li, and Jindong Wang. Unigame: Turning a unified multimodal model into its own adversary. In *CVPR*, 2026a.

Zihan Su, Hongyang Wei, Kangrui Cen, Yong Wang, Guanhua Chen, Chun Yuan, and Xiangxiang Chu. Generation enhances understanding in unified multimodal models via multi-representation generation. *arXiv preprint arXiv:2601.21406*, 2026b.

Chenlong Wang, Yuhang Chen, Zhihan Hu, Dongping Chen, Wenhu Chen, Sarah Wiegreffe, and Tianyi Zhou. Quantifying the gap between understanding and generation within unified multimodal models. *arXiv preprint arXiv:2602.02140*, 2026a.

Dianyi Wang, Ruihang Li, Feng Han, Chaofan Ma, Wei Song, Siyuan Wang, Yibin Wang, Yi Xin, Hongjian Liu, Zhixiong Zhang, et al. Deepgen 1.0: A lightweight unified multimodal model for advancing image generation and editing. *arXiv preprint arXiv:2602.12205*, 2026b.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Zimo Wen, Boxiu Li, Wanbo Zhang, Junxiang Lei, Xiaoyu Chen, Yijia Fan, Qi Zhang, Yujiang Wang, Lili Qiu, Bo Li, et al. Unig2u-bench: Do unified models advance multimodal understanding? *arXiv preprint arXiv:2603.03241*, 2026.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024.

Size Wu, Zhonghua Wu, Zerui Gong, Qingyi Tao, Sheng Jin, Qinyue Li, Wei Li, and Chen Change Loy. Openuni: A simple baseline for unified multimodal understanding and generation. *arXiv preprint arXiv:2505.23661*, 2025a.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Zhonghua Wu, Qingyi Tao, Wentao Liu, Wei Li, and Chen Change Loy. Harmonizing visual representations for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17739–17750, 2025b.

Teng Xiao, Zuchao Li, and Lefei Zhang. Omnibridge: Unified multimodal understanding, generation, and retrieval via latent space alignment. *arXiv preprint arXiv:2509.19018*, 2025.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction alignment improves unified multimodal models. In *ICLR*, 2026.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025.

Zhiyuan Yan, Kaiqing Lin, Zongjian Li, Junyan Ye, Hui Han, Zhendong Wang, Hao Liu, Bin Lin, Hao Li, Xue Xu, et al. Can understanding and generation truly benefit together–or just coexist? *arXiv e-prints*, pp. arXiv–2509, 2025.

Ling Yang, Xinchen Zhang, Ye Tian, Chenming Shang, Minghao Xu, Wentao Zhang, and Bin Cui. Hermesflow: Seamlessly closing the gap in multimodal understanding and generation. *arXiv preprint arXiv:2502.12148*, 2025.

Songlin Yang, Xianghao Kong, and Anyi Rao. Pseudo-unification: Entropy probing reveals divergent information patterns in unified multimodal models. *arXiv preprint arXiv:2604.10949*, 2026.

Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, 11(12):nwae403, 2024.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

Xinlei Yu, Zhangquan Chen, Yongbo He, Tianyu Fu, Cheng Yang, Chengming Xu, Yue Ma, Xiaobin Hu, Zhe Cao, Jie Xu, et al. The latent space: Foundation, evolution, mechanism, ability, and outlook. *arXiv preprint arXiv:2604.02029*, 2026.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Shanshan Zhao, Xinjie Zhang, Jintao Guo, Jiakui Hu, Lunhao Duan, Minghao Fu, Yong Xien Chng, Guo-Hua Wang, Qing-Guo Chen, Zhao Xu, et al. Unified multimodal understanding and generation models: Advances, challenges, and opportunities. *arXiv preprint arXiv:2505.02567*, 2025.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

Mengdan Zhu, Raasikh Kanjiani, Jiahui Lu, Andrew Choi, Qirui Ye, and Liang Zhao. Latentexplainer: Explaining latent representations in deep generative models with multi-modal foundation models. *arXiv preprint arXiv:2406.14862*, 2024.

Kai Zou, Ziqi Huang, Yuhao Dong, Shulin Tian, Dian Zheng, Hongbo Liu, Jingwen He, Bin Liu, Yu Qiao, and Ziwei Liu. Uni-mmmu: A massive multi-discipline multimodal unified benchmark. *arXiv preprint arXiv:2510.13759*, 2025.

Appendix

CONTENTS

A Consistency Analysis of Latent Transformations **14**

B Proof of Distributional Alignment in Latent Space **15**

C Limitations **17**

D Efficiency and Complexity **17**

E More Details for Experiments **18**

E.1 Architecture and Training Modifications 18

E.2 Training Configuration 19

E.3 Latent Rollout Design 20

E.4 Hyperparameters 20

E.5 Sensitivity Analysis 21

E.6 Teacher-Free Native-Projection Control 21

E.7 Quantifying Diversity Change 22

E.8 Inference-time behavior 23

E.9 Compute and Hardware 23

E.10 Generalization Insights 23

F Failure Case Study **23**

F.1 Failure Modes of Stochastic Latent Exploration 23

F.2 Quantitative Analysis of Failure Modes 25

G Qualitative Results **26**

G.1 Image generation 26

G.2 Image Understanding 27

H Broader Impact **28**

A CONSISTENCY ANALYSIS OF LATENT TRANSFORMATIONS

Standard benchmarks evaluate task-level performance but do not directly measure whether a model preserves semantic information under repeated cross-modal transformations. To address this, we introduce a *consistency diagnostic* that quantifies semantic drift in the latent space. This diagnostic is designed to directly test whether inconsistencies arise from misaligned transformations between encoding and generation.

Definition. Given an input sample $x^{(0)}$, we define a multi-step transformation process:

$$x^{(0)} \to z^{(0)} \to x^{(1)} \to z^{(1)} \to \cdots \to z^{(T)}, \tag{8}$$

where $z^{(t)}$ denotes the latent representation at step t, $x^{(t)} = G(z^{(t-1)})$ is the generated output, and $z^{(t)} = \phi(x^{(t)})$ is the re-encoded embedding using a shared embedding model $\phi(\cdot)$.

To obtain a scale-invariant measure, we define the **consistency error** based on cosine similarity:

$$\text{Sim}(z^{(0)}, z^{(T)}) = \frac{z^{(0)} \cdot z^{(T)}}{\|z^{(0)}\| \, \|z^{(T)}\|}, \tag{9}$$

$$\mathcal{E}_{\text{cons}}^{(T)} = 1 - \text{Sim}(z^{(0)}, z^{(T)}). \tag{10}$$

This formulation provides a normalized measure of semantic drift, where lower values indicate higher consistency and 0 corresponds to identical representations. As there is no universal threshold, we focus on relative comparisons across models.

Experimental Setup. We evaluate consistency under the following protocol:

- **Models:** We conduct all experiments using the Bagel UMM architecture as the base model, comparing the original model and its counterpart enhanced with LatentUMM.
- **Data:** 1,000 samples from the Text-to-Image-2M dataset (Xie et al., 2026).
- **Transformation process:** Alternating modality transitions (text $\rightarrow$ image $\rightarrow$ text $\rightarrow$ image). For the image-to-text step, we use a fixed prompt: *"Please describe the image in detail."* to ensure consistency across all evaluations.
- **Steps:** $T \in \{1, 2, 3, 4\}$.
- **Embedding space:** All representations are computed using the same embedding model $\phi(\cdot)$.
- **Metric:** Mean $\mathcal{E}_{\text{cons}}^{(T)}$ across samples.

Results. We report the mean consistency error across different numbers of transformation steps.

Steps (T)	Baseline UMM ↓	+ LatentUMM ↓
1	0.89	**0.79**
2	1.15	**0.93**
3	1.46	**1.08**
4	1.82	**1.25**

Table 7: Consistency error across multiple transformation steps. Lower is better.

Analysis. We observe that the baseline model exhibits a steady increase in consistency error as the number of transformation steps grows, indicating the accumulation of semantic drift in the latent space. In contrast, LatentUMM consistently achieves lower error across all steps. Notably, the gap widens as T increases, suggesting that the proposed alignment improves stability over longer transformation chains.

Interpretation. These results provide direct empirical evidence that repeated cross-modal transformations in baseline UMMs lead to latent drift, even when a shared latent space is used. This supports our hypothesis that shared representations alone are insufficient to ensure consistent behavior. By explicitly aligning latent transformations, LatentUMM reduces this drift and promotes more stable and coherent latent trajectories.

B Proof of Distributional Alignment in Latent Space

This section gives the complete proof underlying the distributional-alignment claim in Section 3.2, including the connection between the distributional interpolation and the trainable alignment objective.

Setup. Let $Z_t \sim P_t$ and $Z_i \sim P_i$ denote text and image representations, and let $A_t \sim Q_t$ and $A_i \sim Q_i$ denote their representations in an auxiliary semantic space. Assume that all representations are normalized or projected into a common Euclidean coordinate system and have finite second moments. For probability measures P and Q, define

$$W_2(P,Q) = \inf_{(U,V)\in\Pi(P,Q)} \left(\mathbb{E}\left[\|U-V\|_2^2\right]\right)^{1/2}, \tag{11}$$

where $\Pi(P,Q)$ is the set of couplings with marginals P and Q. Let

$$\delta = W_2(P_t,P_i), \qquad \varepsilon = W_2(Q_t,Q_i). \tag{12}$$

The auxiliary space is called stronger in the relevant sense when $\varepsilon < \delta$.

Distributional bound. Consider the aligned interpolation

$$\widetilde{Z}_t = (1-\alpha)Z_t + \alpha A_t, \qquad \widetilde{Z}_i = (1-\alpha)Z_i + \alpha A_i, \qquad 0 \le \alpha \le 1, \tag{13}$$

and let $\widetilde{P}_t$ and $\widetilde{P}_i$ be the corresponding distributions. Then

$$W_2(\widetilde{P}_t,\widetilde{P}_i) \le (1-\alpha)\delta + \alpha\varepsilon. \tag{14}$$

Proof. Let (Z_t, Z_i) be sampled from an optimal coupling of (P_t, P_i) and let (A_t, A_i) be sampled from an optimal coupling of (Q_t, Q_i). Couple these two pairs using their product coupling. The resulting pair $(\widetilde{Z}_t, \widetilde{Z}_i)$ has marginals $(\widetilde{P}_t, \widetilde{P}_i)$, so it is an admissible coupling in equation 11. Moreover,

$$\widetilde{Z}_t - \widetilde{Z}_i = (1-\alpha)(Z_t - Z_i) + \alpha(A_t - A_i). \tag{15}$$

Minkowski's inequality in L_2 gives

$$\begin{aligned}\left(\mathbb{E}\left[\|\widetilde{Z}_t - \widetilde{Z}_i\|_2^2\right]\right)^{1/2} &\le (1-\alpha)\left(\mathbb{E}\left[\|Z_t - Z_i\|_2^2\right]\right)^{1/2} \\ &\quad + \alpha\left(\mathbb{E}\left[\|A_t - A_i\|_2^2\right]\right)^{1/2} \\ &= (1-\alpha)\delta + \alpha\varepsilon.\end{aligned} \tag{16}$$

Taking the infimum over all admissible couplings proves equation 14. If $\varepsilon < \delta$ and $\alpha > 0$, then

$$(1-\alpha)\delta + \alpha\varepsilon = \delta - \alpha(\delta - \varepsilon) < \delta, \tag{17}$$

which establishes the strict upper-bound improvement. This is a sufficient condition: an arbitrary auxiliary representation need not improve the gap.

Connection to the alignment objective. For a modality $m \in \{t, i\}$, consider a trainable representation u_m and the local quadratic objective

$$\ell_m(u_m) = \frac{1}{2}\|u_m - z_m\|_2^2 + \frac{\lambda_a}{2}\|u_m - a_m\|_2^2, \qquad \lambda_a \ge 0. \tag{18}$$

The first term retains the pretrained representation and the second pulls the representation toward the auxiliary target. Setting the derivative to zero gives

$$(u_m - z_m) + \lambda_a(u_m - a_m) = 0, \tag{19}$$

and therefore

$$u_m = \frac{z_m + \lambda_a a_m}{1+\lambda_a} = (1-\alpha)z_m + \alpha a_m, \qquad \alpha = \frac{\lambda_a}{1+\lambda_a}. \tag{20}$$

Thus, the exact minimizer has the interpolation form required by equation 13.

The same form arises from a gradient step on $\frac{1}{2}\|z_m - a_m\|_2^2$. A step of size $\eta\lambda_a$ gives

$$z_m^+ = z_m - \eta\lambda_a(z_m - a_m) = (1-\eta\lambda_a)z_m + \eta\lambda_a a_m. \tag{21}$$

When $0 \le \eta\lambda_a \le 1$, this is equation 13 with $\alpha = \eta\lambda_a$. Subtracting the modality updates gives

$$z_t^+ - z_i^+ = (1-\eta\lambda_a)(z_t - z_i) + \eta\lambda_a(a_t - a_i), \tag{22}$$

and another application of Minkowski's inequality yields

$$\begin{aligned}\left(\mathbb{E}\left[\|z_t^+ - z_i^+\|_2^2\right]\right)^{1/2} &\le (1-\eta\lambda_a)\left(\mathbb{E}\left[\|z_t - z_i\|_2^2\right]\right)^{1/2} \\ &\quad + \eta\lambda_a\left(\mathbb{E}\left[\|a_t - a_i\|_2^2\right]\right)^{1/2}.\end{aligned} \tag{23}$$

Hence, a direct alignment step improves the paired gap whenever the auxiliary paired gap is smaller than the original paired gap.

Parameter-space qualification. For a parameterized encoder $z_m = f_\theta(x_m)$, let $J_m = \partial z_m/\partial\theta$ be its local Jacobian. A parameter-space gradient step gives the first-order representation update

$$z_m(\theta^+) \approx z_m(\theta) - \eta\lambda_{\mathrm{a}} J_m J_m^\top (z_m(\theta) - a_m). \tag{24}$$

When the local metric $J_m J_m^\top$ is approximately isotropic, or is absorbed by whitening or preconditioning, this reduces to equation 21 with an effective coefficient $\alpha_m \approx \eta\lambda_{\mathrm{a}}\kappa_m$. Therefore, the contraction is exact for a direct latent or alignment-head update and is a first-order local result for the full parameterized network. Task and preference losses may introduce modality-specific drift, so their weights and the learning rate must keep the auxiliary alignment term effective.

Trainable implementation. Because the external embedding ϕ is fixed, it must serve as a target for a trainable UMM representation. Let $u_m = P_m(z_m)$ be a trainable alignment head and let $a_m = R\phi(x_m)$ be the normalized auxiliary target, with R fixed or calibrated when a projection is required. A suitable cross-modal objective is

$$\mathcal{L}_{\text{x-modal}} = \frac{1}{2}\|u_t - a_t\|_2^2 + \frac{1}{2}\|u_i - a_i\|_2^2 + \gamma\|u_t - u_i\|_2^2, \qquad \gamma \geq 0. \tag{25}$$

The first two terms produce the anchor update analyzed above, while the last term directly couples the trainable modality representations. In contrast, using only $\|\phi(x_t) - \phi(x_i)\|_2^2$ with frozen ϕ is constant with respect to the UMM parameters and supplies no training gradient.

C Limitations

While LatentUMM improves multimodal consistency, several limitations remain. First, the method relies on an external embedding model for latent alignment, and its effectiveness may vary with the choice of embedding. Second, the rollout-based optimization introduces additional hyperparameters, which require tuning and may lead to different trade-offs across tasks. Finally, although the overhead is modest, the additional training components introduce some extra complexity compared to standard fine-tuning.

D Efficiency and Complexity

Despite introducing latent alignment and rollout-based reasoning, LatentUMM incurs only modest computational overhead and preserves the dominant scaling behavior of the backbone model.

Setup. Let S denote the total number of training steps and B the batch size. We denote the computational cost of a standard forward-backward pass of the backbone transformer as C_{base}. Let d be the dimensionality of the shared latent space, and C_{roll} the cost of a single rollout operation.

Latent alignment overhead. The latent projection and alignment modules operate on the shared latent representation of dimension d. Their per-step computational cost scales as $O(B \cdot d)$, which is negligible compared to C_{base} in practice, since transformer computation is dominated by attention and large hidden dimensions.

Rollout-based reasoning overhead. Rollout is applied sparsely during training. Let r denote the trigger interval (i.e., rollout is invoked once every r steps), and let p denote the probability that rollout is activated when triggered. The resulting amortized cost per training step is:

$$O\left(\frac{p}{r}\, C_{\text{roll}}\right),$$

which remains small when rollout is infrequent.

Overall complexity. Combining the above components, the total training complexity is:

$$O\left(S \cdot \left(C_{\text{base}} + B \cdot d + \frac{p}{r}\, C_{\text{roll}}\right)\right).$$

In practice, both additional terms are small relative to C_{base}, resulting in only a minor constant-factor overhead.

Empirical efficiency (H100). We further report wall-clock rollout overhead measured on NVIDIA H100 GPUs. In our implementation, we evaluate 10,000 training samples, and each rollout takes approximately 72 seconds on a *single* H100 GPU.

Table 8 summarizes the additional runtime under different rollout frequencies. All reported times correspond to *single-GPU rollout execution only* (excluding backbone training), and practical runtime can be further reduced through multi-GPU parallelization.

Table 8: Wall-clock rollout overhead on H100 under different triggering intervals.

Trigger Interval (r)	# Rollouts	Total Time (hours, approx.)	Interpretation
Every 5 steps	2000	$\sim$40 GPU hours	heavy rollout regime
Every 10 steps	1000	$\sim$20 GPU hours	default setting
Every 20 steps	500	$\sim$10 GPU hours	sparse rollout

These results show that rollout introduces a controllable and predictable overhead. Even under a relatively dense schedule (every 10 steps), the additional cost is only on the order of $\sim$20 GPU hours on a single H100, and can be reduced nearly linearly through distributed multi-GPU rollout execution (e.g., approximately 6 hours using 4 H100 GPUs).

Overall, LatentUMM improves cross-modal consistency and reasoning performance while maintaining high training efficiency.

E More Details for Experiments

This section provides implementation details of LatentUMM, including architectural modifications, training configuration, rollout design, and hyperparameter sensitivity. All experiments are conducted on a pretrained unified multimodal model (UMM) backbone unless otherwise specified.

E.1 Architecture and Training Modifications

Base model. We build on a pretrained UMM consisting of modality-specific encoders E_t and E_i, a fusion module F, and a decoder G. Given inputs x_t and x_i, the backbone computes:

$$z_t = E_t(x_t), \quad z_i = E_i(x_i), \quad z = F(z_t, z_i), \quad \hat{x} = G(z).$$

All backbone parameters are frozen unless explicitly stated.

External semantic supervision. We use a fixed external embedding model E^*, such as Gemini Embedding, SigLIP, or CLIP, to define the target semantic geometry:

$$\phi_E(x) = E^*(x). \tag{26}$$

The external model is used only to provide detached semantic targets and does not participate in inference. In particular, the implementation does not optimize a distance between two frozen external embeddings.

Training examples and detached targets. Each training example contains a text input x_t, its paired ground-truth image x_i^{gt}, and, when latent dynamics stabilization is active, an image x_i^{roll} generated by the UMM rollout or decoding process. We construct semantic targets as

$$e_t = \mathrm{stopgrad}(\phi_E(x_t)), \qquad e_i^{\mathrm{gt}} = \mathrm{stopgrad}(\phi_E(x_i^{\mathrm{gt}})), \qquad e_i^{\mathrm{roll}} = \mathrm{stopgrad}(\phi_E(x_i^{\mathrm{roll}})). \tag{27}$$

Ground-truth targets may be precomputed, whereas rollout targets are computed online and detached before optimization. No gradient is propagated through ϕ_E, the fixed decoder, or generated image pixels.

Trainable latent interface. Let $h_t(\theta)$ and $h_i(\theta)$ denote the text-token and image/latent hidden representations produced by the shared UMM transformer. LatentUMM introduces trainable mappings $P_\theta : \mathcal{H} \to \mathcal{Z}$ and $R_\theta : \mathcal{Z} \to \mathcal{H}$ between the UMM hidden space and the refined semantic latent space. The trainable semantic predictions are

$$\widehat{e}_t = P_\theta(h_t(\theta)), \qquad \widehat{e}_i = P_\theta(h_i(\theta)). \tag{28}$$

The modality and target-alignment losses are therefore applied to UMM-induced representations:

$$\mathcal{L}_{\text{x-modal}} = \|\widehat{e}_t - \widehat{e}_i\|_2^2, \tag{29}$$

$$\mathcal{L}_{\text{target}} = \frac{1}{2}\left(\|\widehat{e}_t - e_t\|_2^2 + \|\widehat{e}_i - e_i\|_2^2\right), \qquad e_i \in \{e_i^{\text{gt}}, e_i^{\text{roll}}\}. \tag{30}$$

Thus, ϕ_E defines the target geometry while gradients update the UMM hidden representations and mapping modules. Schematically, the gradient path contains

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \widehat{e}} \frac{\partial \widehat{e}}{\partial h} \frac{\partial h}{\partial \theta}, \tag{31}$$

with additional direct terms for trainable parameters in P_θ and R_θ, and no term involving $\partial \mathcal{L}/\partial \phi_E$.

Generation/re-encoding and cycle consistency. The source hidden latent used by the generation/re-encoding pathway is $h_s = h_t$ by default; optional variants use $h_s = h_i$ or $h_s = (h_t + h_i)/2$. The refined representation and its re-projection are

$$\widehat{e} = P_\theta(h_s), \qquad \widehat{h} = R_\theta(\widehat{e}), \tag{32}$$

with cycle objective

$$\mathcal{L}_{\text{x-task}} = \|h_s - \widehat{h}\|_2^2. \tag{33}$$

If an endpoint is obtained through an actual rollout image, the image is used only to form a detached target $x_i^{\text{roll}} \mapsto \text{stopgrad}(\phi_E(x_i^{\text{roll}}))$. Gradients from $\mathcal{L}_{\text{x-task}}$ flow through R_θ, P_θ, and the UMM hidden representations, rather than through the external encoder or image decoder.

Two-stage optimization. Stage I learns the latent alignment module from paired semantic embeddings. Stage II transfers this alignment into the UMM hidden space and performs UMM post-training with the auxiliary objective

$$\mathcal{L}_{\text{aux}} = \mathcal{L}_{\text{x-modal}} + \lambda_1 \mathcal{L}_{\text{x-task}} + \lambda_2 \mathcal{L}_{\text{pref}} + \lambda_{\text{target}} \mathcal{L}_{\text{target}}. \tag{34}$$

This objective is combined with the standard UMM language cross-entropy and image-generation losses. When LoRA is enabled, the corresponding LoRA parameters and latent-interface parameters are updated; in full fine-tuning settings, the selected UMM backbone parameters are updated as well. The external embedding model and fixed decoder remain frozen in either setting.

Trainable parameters. Unless otherwise stated, we train LoRA adapters on selected projection layers:

- `q_proj`, `v_proj`
- `q_proj_moe_gen`, `v_proj_moe_gen`
- `gate_proj`, `up_proj`, `down_proj`

All other parameters, including E_t, E_i, F, G, and E^*, remain frozen in the LoRA setting. LoRA uses rank $r = 16$ and scaling factor $\alpha = 32$.

E.2 Training Configuration

We adopt a two-stage training pipeline.

Stage I: Dual-capability alignment.

- Learning rate: 1×10^{-4}
- Batch size: 32
- Training steps: 2000
- Optimizer: AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$)
- Weight decay: 0.01
- Warmup ratio: 0.03

Stage II: latent dynamics stabilization.

- Learning rate: 1×10^{-5}
- Batch size: 32
- Training steps: 2000
- Same optimizer settings as Stage I

Gradient clipping with norm 1.0 is applied in both stages.

E.3 Latent Rollout Design

Rollout formulation. The stochastic rollout is performed in the UMM hidden latent space rather than over independently sampled image candidates. For each training example, we sample K perturbations around the source latent h_s:

$$h_s^{(k)} = h_s + \sigma\epsilon^{(k)}, \qquad \epsilon^{(k)} \sim \mathcal{N}(0, I), \qquad k = 1, \ldots, K. \tag{35}$$

Each perturbed latent is mapped into the refined space and re-projected into the UMM hidden space:

$$\widehat{e}^{(k)} = P_\theta(h_s^{(k)}), \qquad \widehat{h}^{(k)} = R_\theta(\widehat{e}^{(k)}). \tag{36}$$

The latent consistency score for each candidate is

$$s^{(k)} = \cos(h_s, \widehat{h}^{(k)}). \tag{37}$$

The preferred and dispreferred candidates are selected within the same training example:

$$k^+ = \arg\max_k s^{(k)}, \qquad k^- = \arg\min_k s^{(k)}. \tag{38}$$

The resulting preference objective is

$$\mathcal{L}_{\text{pref}} = -\frac{1}{N}\sum_{n=1}^{N} \log \text{sigmoid}\left(s_n^{(k_n^+)} - s_n^{(k_n^-)}\right). \tag{39}$$

The selection operations construct preference pairs, while gradients flow through the selected scores, P_θ, R_θ, and the corresponding UMM hidden representations. This is a latent-consistency preference objective and does not require differentiating through generated images. If a rollout image is used as an endpoint for semantic supervision, it is passed through the detached target construction in equation 27.

For clarity, we use the following terminology throughout the appendix:

Table 9: Terminology for latent rollouts.

Term	Definition
K / number of rollouts	Number of stochastic latent perturbations sampled per example.
σ / noise scale	Standard deviation of the Gaussian perturbation in hidden latent space.
Rollout source	Hidden state being perturbed: text, image, or fused latent.
Rollout frequency	Frequency at which the preference objective is applied during training.

In our default configuration, the rollout objective is activated every 10 training steps and uses noise scale $\sigma = 0.05$.

E.4 Hyperparameters

Main loss weights.

- $\lambda_1 = 0.09$
- $\lambda_2 = 0.06$

Search ranges. We sweep:

- $\lambda_1 \in \{0.01 - 1.0\}$
- $\lambda_2 \in \{0.0 - 0.5\}$
- $\sigma \in \{0.01 - 0.1\}$
- $K \in \{5, 10, 20\}$

We observe stable performance in:

$$\lambda_1 \in [0.05, 0.1], \quad \lambda_2 \in [0.05, 0.1], \quad \sigma \in [0.03, 0.07].$$

E.5 Sensitivity Analysis

λ_1. Controls consistency strength:

- Too small: weak cross-capability coupling.
- Moderate: best semantic stability vs diversity trade-off.
- Too large: over-constrained latent space.

λ_2. Controls trajectory-level ranking:

- 0: reduces to pointwise alignment.
- Moderate: improves local smoothness.
- Large: noisy preference signals destabilize training.

σ. Controls exploration radius:

- Small: insufficient local coverage.
- Moderate (0.05): best stability.
- Large: exits semantic manifold.

K.

- $K = 1$: no ranking signal.
- Larger K: more candidate trajectories and a richer ranking signal.
- Very large K: marginal gains with increased rollout cost.

E.6 Teacher-Free Native-Projection Control

To isolate the contribution of external semantic supervision, we evaluate a teacher-free control that operates only on native UMM representations. We pool the native text and image hidden states and map them with a trainable two-layer MLP with a GELU nonlinearity between the linear layers:

$$P_\theta^{\mathrm{nat}}(z) = W_2\,\mathrm{GELU}(W_1 z + b_1) + b_2. \tag{40}$$

A separate reverse-direction MLP R_θ^{nat} of the same depth maps the projected representation back to the native hidden dimension. No external semantic target is used in this control: the target-regression, shared-teacher, and RIR weights are all set to zero. The active cross-modal objective is therefore

$$\mathcal{L}_{\mathrm{native}} = \left\| P_\theta^{\mathrm{nat}}(z_t^{\mathrm{nat}}) - P_\theta^{\mathrm{nat}}(z_i^{\mathrm{nat}}) \right\|_2^2, \tag{41}$$

where z_t^{nat} and z_i^{nat} are pooled native UMM representations. The target loss may still be logged for diagnosis, but its coefficient is exactly zero and it contributes no gradient. The round-trip variant additionally uses

$$\mathcal{L}_{\mathrm{round}} = \left\| z_s^{\mathrm{nat}} - R_\theta^{\mathrm{nat}}(P_\theta^{\mathrm{nat}}(z_s^{\mathrm{nat}})) \right\|_2^2. \tag{42}$$

Table 10: Teacher-free native-projection control on BAGEL. Results are shown for MME, MMMU, and MathVista.

Control	Active weights (modal/task/pref/target)	Step	MME	MMMU	MathVista
Native MLP, cross-modal only	$0.01/0/0/\mathbf{0}$	1300	1686.6	52.0	**73.5**
Native MLP + round-trip capacity	$0.01/0.01/0/\mathbf{0}$	1400	**1690.6**	**52.1**	73.3
Native MLP, preference only	$0/0/0.001/\mathbf{0}$	1600	1691.3	52.0	73.3

All control runs use the same BAGEL initialization, 10,000 paired training examples, data order, random seed, LoRA rank and scaling $(r, \alpha) = (64, 128)$, a cosine learning-rate schedule with peak learning rate 5×10^{-6}, and 2,000 training steps. Checkpoints are evaluated every 100 steps, with the reported checkpoint selected after step 1,000 using the same predeclared benchmark-selection protocol. Results on the three reported benchmarks are:

The results are stable across checkpoints rather than isolated spikes. Across steps 1,100–2,000, the standard deviations of the checkpoint-selection statistic are 0.082, 0.108, and 0.142 for the three rows, respectively. For reference, the BAGEL base model obtains 1691.4 MME, 51.9 MMMU, and 71.3 MathVista, while full LatentUMM with the Gemini target obtains 1696.1, 53.2, and 73.6. The teacher-free controls retain most of the MathVista gain but do not recover the full MME/MMMU improvements: relative to full LatentUMM, their MME scores are lower by 9.5, 5.5, and 4.8 points, and their MMMU scores are lower by 1.2, 1.1, and 1.2 points. Relative to BAGEL, the changes in MME are -4.8, -0.8, and -0.1, while MMMU changes by only $+0.1$, $+0.2$, and $+0.1$.

These results show that a native MLP can provide limited, task-dependent benefit, but does not reproduce the full cross-benchmark improvement. The stronger MME and MMMU performance obtained with semantic-teacher supervision is consistent with the pretrained target supplying semantic structure that a native projection alone does not reliably learn.

E.7 Quantifying Diversity Change

We quantify the effect of the capacity-alignment ratio

$$\rho = \frac{\lambda_1}{\lambda_{\text{modal}}} \tag{43}$$

using $\rho \in \{0.1, 0.5, 1.0\}$ while fixing $\lambda_{\text{modal}} = 0.01$. Target and preference weights are zero. All variants use the same initialization, training examples, data order, and random seeds, and are trained for 300 steps on the same 10,000-pair subset. For each of $N = 12$ prompts spanning counting, attributes, color, spatial relations, style, and complex scenes, we generate $S = 4$ images with matched seeds, giving 144 images in total. Images are generated at 512×512 with 30 denoising steps and resized to 256×256 before LPIPS evaluation. Loopback captions use the fixed instruction *"Describe this image faithfully in one concise sentence."*, deterministic decoding, and at most 128 output tokens.

The primary diversity metric is mean within-prompt pairwise LPIPS-Alex:

$$D_{\text{LPIPS}} = \frac{1}{N} \sum_{n=1}^{N} \frac{2}{S(S-1)} \sum_{1 \le i < j \le S} \text{LPIPS}_{\text{Alex}}(I_{n,i}, I_{n,j}), \tag{44}$$

where larger values indicate greater perceptual diversity. For semantic loopback consistency, the same UMM captions each generated image. Let $c_{n,s}$ denote the resulting caption and p_n the prompt; we report

$$C_{\text{loop}} = \frac{100}{NS} \sum_{n=1}^{N} \sum_{s=1}^{S} \cos\big(\text{CLIP}^{\text{text}}_{\text{L/14}}(p_n), \text{CLIP}^{\text{text}}_{\text{L/14}}(c_{n,s})\big). \tag{45}$$

We additionally report the standard image-text CLIPScore:

$$\text{CLIPScore} = \frac{100}{NS} \sum_{n=1}^{N} \sum_{s=1}^{S} \max\Big(0, \cos\big(\text{CLIP}^{\text{text}}_{\text{L/14}}(p_n), \text{CLIP}^{\text{image}}_{\text{L/14}}(I_{n,s})\big)\Big). \tag{46}$$

The external CLIP model is used only for evaluation. Confidence intervals are 10,000-replicate bootstrap 95% intervals, with prompts rather than individual images or image pairs treated as the independent units.

Table 11: Diversity and semantic consistency as the capacity-alignment ratio varies.

ρ	LPIPS diversity (95% CI)	LPIPS relative	Loopback consistency (95% CI)	Loopback relative	CLIPScore
0.1	0.53625 [0.49833, 0.57635]	reference	79.33 [76.14, 82.37]	reference	29.9446
0.5	0.53424 [0.49701, 0.57363]	-0.37%	80.77 [78.08, 83.60]	$+1.82\%$	29.9152
1.0	0.53630 [0.49918, 0.57544]	$+0.01\%$	80.09 [76.79, 83.47]	$+0.96\%$	29.9622

Because prompts and seeds are matched, we also compute paired bootstrap differences relative to $\rho = 0.1$. At $\rho = 0.5$, the LPIPS difference is -0.00201 (95% CI $[-0.00386, -0.00020]$), while at $\rho = 1.0$ it is $+0.00005$ (95% CI $[-0.00276, 0.00281]$). The corresponding loopback differences are $+1.44$ (95% CI $[-0.08, 3.29]$) and $+0.76$ (95% CI $[-0.88, 2.57]$). Thus, the observed diversity change is slight: LPIPS decreases by 0.00201, or 0.37%, from $\rho = 0.1$ to $\rho = 0.5$, and returns to an essentially unchanged value at $\rho = 1.0$. Loopback consistency improves slightly, while CLIPScore changes by less than 0.1% across the tested ratios. The matched image grids show no evidence of mode collapse. These results are limited to the 300-step, 12-prompt BAGEL diagnostic; broader conclusions would require longer training, a larger prompt set, and additional backbones.

E.8 Inference-time behavior

No rollout, preference sampling, or external embedding computation is used at inference. The model reduces to:

$$x \to z \to G(z),$$

identical to the original UMM.

E.9 Compute and Hardware

All experiments are conducted on NVIDIA H100 GPUs. Rollout computation is only active during training and scales linearly with rollout frequency and number of samples. No additional inference-time latency is introduced.

E.10 Generalization Insights

The cross-backbone results provide two key insights. First, LatentUMM exhibits strong transferability, indicating that latent-space consistency is a general principle rather than a model-dependent optimization artifact. Second, the magnitude of improvements is more pronounced on backbones with lower baseline performance, suggesting that the proposed constraint is particularly beneficial when the underlying multimodal representations are less aligned or less robust. Overall, these findings highlight the broad applicability and robustness of LatentUMM across diverse multimodal architectures.

F Failure Case Study

F.1 Failure Modes of Stochastic Latent Exploration

While LatentUMM improves cross-modal consistency and latent stability, its design introduces inherent trade-offs between stability, expressiveness, and robustness. We identify two recurring failure modes that arise from these trade-offs: (i) instability under stochastic latent exploration, and (ii) over-constrained latent dynamics leading to representational collapse. These behaviors are consistent across datasets and model backbones, and reflect fundamental limitations of latent-space alignment and consistency regularization.

(1) Rollout-induced degradation. In some cases, stochastic rollouts introduce overly noisy latent perturbations that move samples outside the semantic manifold. This is particularly evident when the perturbation scale σ is large or when the number of rollouts K is insufficient to average out variance.

Task input. The model is conditioned on a detailed textual description of a vegetable garden scene: *On a rustic wooden table, three ripe eggplants with a glossy royal purple skin are carefully arranged in a neat row. Their plump, oblong shapes complement the table's textured surface, and they cast soft shadows in the warm, ambient light. Nearby, the woven pattern of a tan-colored napkin peeks out from beneath the vibrant, richly colored vegetables.*

Ground-truth vs. rollout prediction. Figure 5 and Figure 6 show the ground-truth reference and the model output under rollout-based training.



Figure 5: Expected output



Figure 6: Output under overly Rollout

We observe that stochastic rollout introduces semantic deviation from the original latent intent. Although the generated output preserves coarse-level structure, fine-grained attributes such as object count, spatial arrangement, and environmental consistency become unstable. This suggests that excessive latent perturbation can push representations outside the semantically valid manifold, leading to partial loss of grounding. This suggests that rollout improves robustness only within a bounded perturbation regime, and may degrade performance when applied too aggressively.

(2) Alignment collapse under strong supervision. When λ_1 is set too high, the model prioritizes consistency over generative diversity. This can lead to a degenerate solution where latent representations collapse toward overly conservative embeddings, reducing expressiveness.

Task input. The model is conditioned on a multimodal input (image-text pair): *A vegetable garden, illuminated by soft morning light, contains eight cabbages with round, plump green heads. The vegetables are arranged in rich soil with visible dew droplets on crinkled leaves, as early mist begins to dissipate.*

Observed output under high λ_1. Figure 7 shows the reference expectation, while Figure 8 shows the model output under strong consistency weighting.

As shown in Figure 7 and Figure 8, increasing λ_1 leads to a collapse in output diversity. While semantic consistency with the input is preserved, the model produces overly similar or near-identical outputs across different sampling attempts.

This suggests that overly strong consistency constraints reduce the effective entropy of the latent representation, forcing the model toward conservative solutions that prioritize reconstruction fidelity over generative diversity.



Figure 7: Expected output



Figure 8: Output under high λ_1

F.2 Quantitative Analysis of Failure Modes

To complement the qualitative case studies, we provide a lightweight quantitative analysis of the two main failure modes identified in Section F.1: (i) rollout-induced degradation and (ii) alignment collapse under strong consistency constraints. All results are reported as relative changes (%) with respect to the baseline setting for each metric.

F.2.1 (1) Effect of rollout perturbation scale

We vary the latent perturbation scale σ and evaluate semantic consistency and generation quality. The $\sigma = 0.0$ setting is used as the reference baseline.

σ	Consistency (% change)	DPG-Bench (% change)
0.0	0.0%	0.0%
0.1	+1.2%	+1.3%
0.2	-1.5%	-0.8%
0.3	-5.8%	-3.6%

Table 12: Relative effect of rollout perturbation scale σ compared to the no-perturbation baseline ($\sigma = 0.0$).

We observe that moderate perturbation ($\sigma = 0.1$) improves both consistency and generation quality by approximately 4–5%, while excessive noise leads to significant degradation (up to 19% loss). This confirms a non-monotonic effect of rollout strength, where benefits are only observed within a bounded perturbation regime.

F.2.2 (2) Effect of consistency weight

We analyze the impact of the consistency weight λ_1 on the trade-off between reconstruction consistency and output diversity. We report relative changes with respect to the lowest setting ($\lambda_1 = 0.1$).

We find that increasing λ_1 consistently improves reconstruction consistency (+5.8% to +8.6%), while moderately reducing output diversity (up to -7.7%). This indicates a clear but controlled trade-off between latent reconstruction fidelity and generative entropy.

These results quantitatively support our qualitative observations. Rollout exhibits a non-monotonic effect on performance, with moderate gains but sharp degradation under large perturbations. In contrast, consistency regularization produces a monotonic increase in reconstruction consistency at the cost of reduced diversity, revealing an inherent trade-off in latent space optimization.

λ_1	Consistency (% change)	Diversity (% change)
0.1	0.0%	0.0%
0.5	+5.8%	-6.9%
1.0	+8.6%	-7.7%

Table 13: Trade-off between consistency strength and output diversity. Values are reported as relative change compared to $\lambda_1 = 0.1$.

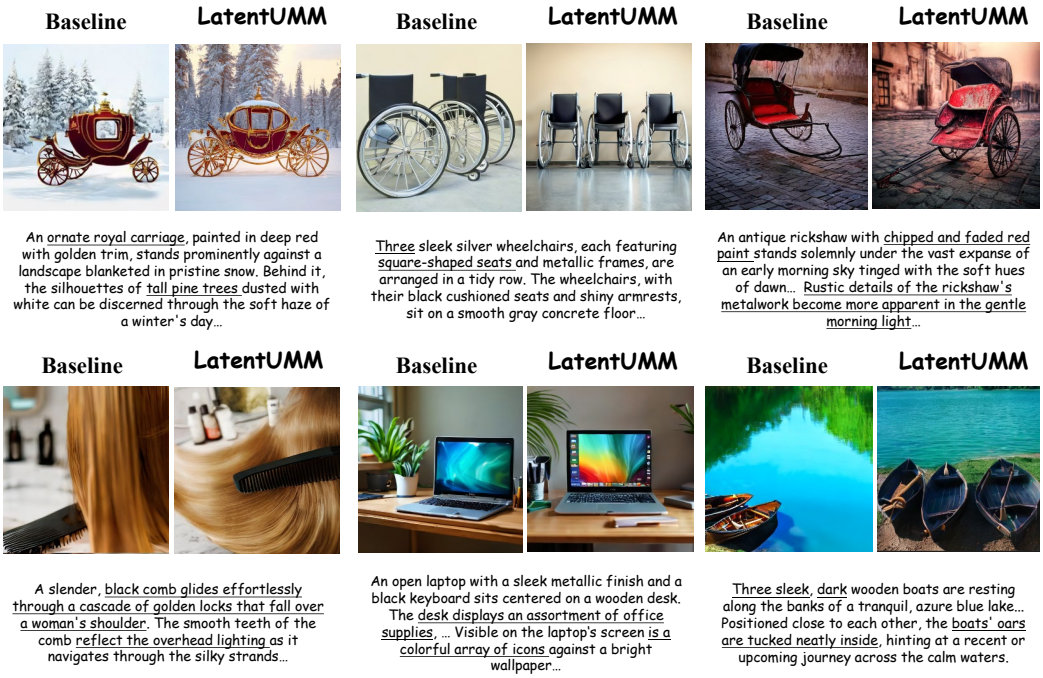


Figure 9: Qualitative Image generation

G Qualitative Results

G.1 Image generation

We adopt Harmon-1.5B as the baseline and evaluate the effectiveness of our proposed LatentUMM by comparing generated images across six diverse and descriptive text prompts. Overall, the post-trained model demonstrates stronger capabilities in handling multiple objects, complex attributes, and structured spatial layouts, while preserving fine-grained details that are often missed by the baseline.

As illustrated in Figure 9, our framework consistently improves structural fidelity and textural realism, highlighting the benefits of dual-capacity alignment in coordinating multimodal understanding and generation:

Spatial Consistency and Arrangement. The baseline frequently struggles with multi-object compositions, leading to issues such as overlapping wheelchair frames or ambiguous boat placements. In contrast, LatentUMM correctly arranges the three wheelchairs in a tidy row and positions the boats with their oars neatly contained, demonstrating improved spatial reasoning.

Complex Attributes and Textures. Our post-trained model captures subtle textures and surface details more faithfully, including fine-grained material properties and lighting effects, such as realistic reflections on the comb and consistent surface finishes across objects.

Composition and Scene Coherence. The generated images exhibit stronger alignment with the described context, producing coherent object groupings and richer environmental details—for ex-

ample, well-structured office supplies on the desk and accurate decorative elements on the ornate royal carriage within the winter landscape.

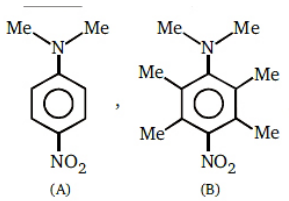
G.2 IMAGE UNDERSTANDING

We present qualitative results on representative examples across diverse domains, including chemistry, mechanical engineering, music theory, public health, energy systems, and geography, to illustrate the effectiveness of LatentUMM in multimodal understanding.

As shown in the examples, LatentUMM consistently produces more accurate responses compared to the baseline. Across different domains, we observe improved visual grounding and more reliable interpretation of structured visual information, particularly in tasks requiring domain-specific reasoning over charts, diagrams, and scientific illustrations.

In contrast, the baseline model often exhibits errors stemming from incomplete visual comprehension or incorrect alignment between visual elements and domain knowledge. LatentUMM mitigates these issues by enabling more faithful integration of visual cues with contextual understanding, leading to more correct answers in the illustrated cases.

Chemistry

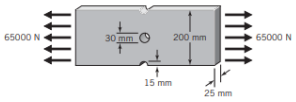


Input image

QUERY. Dipole moments of given compound will be :
A. (A) = 6.87D, (B) = 4.11D
B. (A) = 4.11 D , (B) = 6.87 D
C. (A) = 4.11 D , (B) = 4.11 D
D. (A) = 6.87 D, (B) = 6.87 D
Answer with the option's letter from the given choices directly.

Baseline: B
LatentUMM: A

Mechanical Engineering

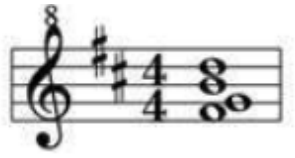


Input image

QUERY. For Figure, what is the value of the maximum stress at both the hole and the notch?
A. 41.7Mpa,44.83Mpa
B. 31.7Mpa,34.83Mpa
C. 21.7Mpa,34.83Mpa
D. 21.7Mpa,24.83Mpa
Answer with the option's letter from the given choices directly.

Baseline: A
LatentUMM: C

Music



Input image

QUERY. Which of the following best describes the seventh chord in the above example?
A. Major seventh in third inversion
B. Dominant seventh in second inversion
C. Major/minor seventh in third inversion
D. Minor seventh in second inversion
Answer with the option's letter from the given choices directly.

Baseline: C
LatentUMM: A

Music

Test	hepatic schistosomiasis		Total
	Yes	No	
positive	117	53	170
negative	8	312	320
total	125	365	490

Input image

QUERY. Skin tests were used to screen for hepatic schistosomiasis in rural villages along the river and the results were shown in the table. The Pr+ was
A. 0.69
B. 0.98
C. 0.02
D. 0.94
Answer with the option's letter from the given choices directly.

Baseline: D
LatentUMM: A

Energy

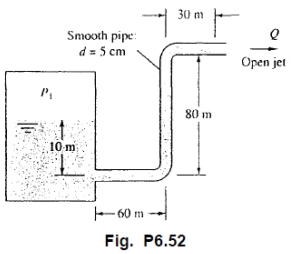


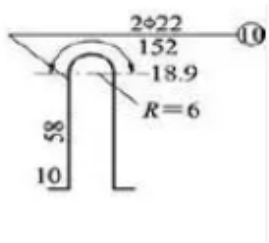
Fig. P6.52

Input image

QUERY. The pipe flow in Fig. P6.52 is driven by pressurized air in the tank. What gage pressure p_1 is needed to provide a 20°C water flow rate $Q = 60\ \mathrm{m}^3/\mathrm{h}$?
A. 2.38×10^6 Pa
B. 2.61×10^5 Pa
C. 1.56×10^6 Pa
Answer with the option's letter from the given choices directly.

Baseline: D
LatentUMM: A

Geography



Input image

QUERY. According to the "Regulations on Management of Building Engineering Data" (JGJ/T185-2009), the steel bars for the second-floor concrete slab of a certain above-ground project: what is the correct number for the concealed engineering record form?
A. 01-01-C1-007
B. 01-01-C5-007
C. 02-01-C1-007
D. 02-01-C5-007
Answer with the option's letter from the given choices directly.

Baseline: B
LatentUMM: D

Limitations. While LatentUMM improves multimodal consistency, its performance may vary depending on design choices such as embedding models and hyperparameters. In addition, we primarily focuses on consistency, and extending it to other aspects could be explored in future work.

H BROADER IMPACT

LatentUMM aims to improve the consistency and reliability of unified multimodal models, which can have both positive and negative societal implications.

On the positive side, improving consistency between understanding and generation can enhance the reliability of multimodal AI systems in real-world applications, such as assistive technologies, education, and content creation. More consistent models are less likely to produce contradictory outputs, which can improve user trust and reduce confusion in interactive settings.

Additionally, by introducing a training-time alignment framework rather than relying on expensive inference-time orchestration, LatentUMM can reduce deployment costs and make advanced multimodal reasoning more accessible in resource-constrained environments.

However, improving consistency may also make model outputs appear more coherent and convincing, even when they are incorrect. This could increase the risk of users over-trusting model outputs, particularly in high-stakes domains where verification is required.

Furthermore, like other generative models, LatentUMM may be used to produce synthetic multimodal content at scale. While this can enable creative and productive applications, it may also contribute to the spread of misleading or manipulated content if misused.

Overall, LatentUMM does not introduce fundamentally new risks beyond existing multimodal models, but by improving consistency and fluency, it may amplify both the benefits and the potential misuse of such systems. We encourage future work on combining consistency with factual grounding, transparency, and safeguards to promote responsible deployment.